Wind Farm Power Generation Control via Double-Network-Based Deep Reinforcement Learning

Jingjie Xie, Hongyang Dong, Xiaowei Zhao, and Aris Karcanias

***Abstract*—A model-free deep reinforcement learning (DRL) method is proposed in this paper to maximize the total power generation of wind farms through the combination of induction control and yaw control. Specifically, a novel double-network-based DRL approach is designed to generate control policies for thrust coefficients and yaw angles simultaneously and separately. Two sets of critic-actor networks are constructed to this end. They are linked by a central power-related reward, providing a coordinated control structure while inheriting the critic-actor mechanism's advantages. Compared with conventional DRL methods, the proposed double-network-based DRL strategy can adapt to the distinctive and incompatible features of different control inputs, guaranteeing a reliable training process and ensuring superior performance. Also, the prioritized experience replay strategy is utilized to improve the training efficiency of deep neural networks. Simulation tests based on a dynamic wind farm simulator show that the proposed method can significantly increase the power generation for wind farms with different layouts.**

***Index Terms*—Reinforcement learning, wind farm control, power generation control, model-free control.**

I. Introduction

WIND energy has received worldwide attention for decades due to its renewable, clean, and sustainable features with extensive research e.g. on wind speed forecasting [1], energy generation [2], [3], and energy storage [4]. This paper focuses on the wind-farm power generation maximization. Traditional generation control strategy for wind farms focuses on maximizing the power output of each individual wind turbine in the farm, which is commonly mentioned as the greedy strategy in the literature [5], [6], [7]. However, the greedy strategy is not optimal when considering the total power production of the whole farm. This is because the aerodynamic interactions among wind turbines can influence the power generation process of each other [8]. The wakes induced by the upstream turbines can reduce downstream turbines' power outputs, decreasing the whole farm's total power production [9]. Therefore, designing centralized wind farm control strategies that consider all turbines together instead of operating each one greedily is vital for the whole farm's generation maximization.

This work has received funding from the UK Engineering and Physical Sciences Research Council (grant number: EP/S000747/1). J. Xie, H. Dong and X. Zhao (Corresponding Author) are with the Intelligent Control & Smart Energy (ICSE) Research Group, School of Engineering, University of Warwick, Coventry CV4 7AL, UK. Emails: jingjie.xie@warwick.ac.uk, hongyang.dong@warwick.ac.uk, xiaowei.zhao@warwick.ac.uk. Aris Karcanias is with PA Consulting, 10 Bressenden Place, London SW1E 5DN, U.K. Email: aris.karcanias@paconsulting.com.

Yaw control and induction control are two typical methods to achieve wind-farm power generation maximization [10], [11]. Specifically, the yaw control tunes turbines' yaw angles to steer wake directions and therefore increase downstream turbines' power outputs. The induction control can also mitigate wake effects and enhance generation efficiency by adjusting the turbines' axial induction factors or alternatively controlling power/thrust coefficients. A yaw-angle optimization strategy was introduced in [12] for improving the power gains. An induction control method was described in [13] by taking the induction factor as the control variable. Ref. [14] proposed a model predictive control strategy to adjust induction factors and performed high-fidelity large-eddy simulations for wind farms. A further study using this method was presented in Ref. [15]. Moreover, an adjoint-based model predictive control scheme was proposed in Ref. [16] to maximize the farm's power production via induction control. However, these methods require analytical and accurate wind farm models. Such model-based approaches are sensitive to modelling errors and uncertainties, which may lead to significantly degraded control performance in practical uses.

Some model-free wind farm control strategies [17], [18], [19], [20], [21] have been proposed to overcome the limitations of model-based approaches. In Ref. [17], a gradient estimation-based approach was proposed to minimize the power loss of wind farms. A game-theory optimization algorithm was applied in Ref. [18] to maximize the farm's total power generation, and another data-driven wind farm control approach toward this end was presented in Ref. [19]. Other model-free control schemes like the random search algorithm [20] and stochastic approximation [21] were also employed to achieve wind-farm power generation maximization. However, these methods still suffer from challenges of requiring large-scale unconstrained searching process and/or lacking adaptability to uncertain environmental conditions.

As a cutting-edge model-free control method, reinforcement learning (RL) is a promising new technology which can address the above challenges. RL has strong learning abilities in finding the optimal control policy by evaluating agents' input and output data. It has been applied to many complex systems, such as robots [22], autonomous vehicles [23] and spacecraft [24]. Several RL schemes [25], [26], [27] have been designed for wind farm control problems. For example, an RL-based distributed control method was introduced in [25] to increase the power output of a wind farm via yaw control, and a Q-learning based algorithm was provided in

[26] for same purposes. Recently, a knowledge-assisted deep deterministic policy gradient algorithm was proposed in Ref. [27] to increase the total power production. However, these elegant results considered either yaw control or induction control while combing both may be able to further enhance the wind-farm power generation.

The challenge in combining the yaw control and the induction control together comes from the incompatibility of these two types of control inputs. On the one hand, yaw actuators usually have large time constants and cannot tolerate rapid yaw changes. On the other hand, induction control usually has a much shorter response time than yaw control and can adapt to irregular and rapid variations. Such a contradictory situation can lead to a long learning process and a bad RL performance if an integrated control sequence is employed (as in standard RL methods). This challenge is addressed in our study.

A novel double-network (DN)-based deep reinforcement learning algorithm is proposed in this paper for the power generation maximization of wind farms by generating control policies for thrust coefficients and yaw angles simultaneously. This method is built upon a state-of-the-art deep RL (DRL) alrorithm named deep deterministic policy gradient (DDPG) [28]. Distinct from the standard DDPG, our DN-DDPG method employs two sets of critic-actor networks to evaluate induction and yaw control policies separately. Also, the prioritized experience replay (PER) strategy [29] is utilized to sample the transitions for deep neural network (DNN) training. Therein, the temporal-difference errors (TD-errors) are used to observe transitions' priorities to improve the training efficiency of the algorithm. Finally, a control-oriented dynamical wind farm simulator (WFSim) [30] is employed to validate the effectiveness of our DN-DDPG algorithm. The main novelties and contributions of this paper are as follows:

(1) A novel DN-DDPG method is proposed to achieve the power generation maximization of wind farms. Distinct from DDPG, the proposed DN-DDPG method employs two sets of critic-actor networks to achieve the combination of induction control and yaw control. It is capable of generating thrust coefficient and yaw angle policies simultaneously and separately via the double critic-actor framework. The double DNNs are linked by a central reward. Such a special structure enables our DN-DDPG to handle the incompatibility between different control signals. This leads to a more reliable training process and superior performance over the standard DDPG method.

(2) The mainstream wind farm control schemes commonly require accurate analytical wind farm dynamics, which may result in significantly degraded control performance due to the inevitable modelling errors and uncertainties. In contrast, the proposed DN-DDPG method is data-driven and model-free. It is insensitive to modeling errors and uncertainties, leading to enhanced adaptability and robustness. Test results with WFSim verifies this fact and show that our DN-DDPG has better performance than the conventional greedy strategy, the DDPG, and the nonlinear model predictive control method.

The remainder of this paper is as follows. The wind farm control problem is described in Section II. The DDPG method and the proposed DN-DDPG method with PER scheme are presented in section III. After that, numerical simulations based on WFSim are demonstrated in Section IV. Finally, conclusive remarks are given in Section V.

II. Problem Formulation

The wind-farm power generation maximization problem is formulated in this section.

A. Power Generation Analysis

The force and power output of each turbine in the farm are respectively defined by [19], [31]

$$F_i = \frac{1}{2}\rho A_d (U_i \cos(\phi_i))^2 C'_{T_i}(a_i, \phi_i) \tag{1}$$

$$P_i = \frac{1}{2}\rho A_d (U_i \cos(\phi_i))^3 C_{P_i}(a_i, \phi_i) \tag{2}$$

where $i = 1, 2, ..., N$. Here N is the total number of turbines. A_d is the area of rotor plane, U_i is the wind speed at the i-th turbine. C'_{T_i} and C_{P_i} are called the thrust and power coefficients of the i-th turbine, respectively. They are related to the axial induction factor a_i and the yaw angle ϕ_i, satisfying $C'_{T_i} = 4a_i/(\cos(\mu\phi_i) - a_i)$ and $C_{P_i} = 4a_i/(\cos(\mu\phi_i) - a_i)^2$, where μ is a constant model parameter. From (1) and (2), one can see that the total power output of the whole farm, formalized by $P_{all} = \sum_{i=1}^{N} P_i$, can be controlled by adjusting ϕ_i and C'_{T_i} of each turbine i in the farm.

As mentioned in the Introduction section, the power extraction process of an upstream turbine j will lead to a wake. This wake can influence the power generation of a downstream turbine i (e.g. the wake induced by the turbine j can change the wind speed at the turbine i, i.e. U_i). In other word, the power output of turbine i is not only decided by its own control policies but also influenced by the control policies of all its upstream turbines. Such complicated aerodynamic interactions and control-strategy couplings make it hard to achieve wind-farm power generation maximization via conventional control methods. A deep RL-based method is proposed in this paper to address this challenge. First, we formulize our control problem in the next subsection.

B. Problem Formulation

This work aims to increase the power generation of wind farms by controlling the yaw angle ϕ_i and thrust coefficient C'_{T_i} of each turbine i in the farm. This task can be considered as an optimal control problem with respect to a long-term reward function $J(t) = \sum_t^{\infty} \gamma_t P_{all}(x(t))$, where t denotes the current time step, $\gamma_t = \gamma^t$ with γ is a discount factor, and the control variable is $x(t) = [C'_{T_1}(t), ..., C'_{T_N}(t), \phi_1(t), ..., \phi_N(t)]^{\mathrm{T}}$. The objective is to find an optimal control policy $x^*(t)$ such that $J(t)$ can be maximized, formulized as

$$x^*(t) = \arg\max \sum_{t}^{\infty} \gamma_t P_{all}(x(t)) \tag{3}$$

subject to

$$x(t) = [C'_{T_1}(t), ..., C'_{T_N}(t), \phi_1(t), ..., \phi_N(t)]^{\mathrm{T}} \tag{4}$$

$$C'_{T,min} \leq C'_{T_i}(t) \leq C'_{T,max} \tag{5}$$

$$\phi_{min} \leq \phi_i(t) \leq \phi_{max} \tag{6}$$

where (5) and (6) represent the constraints of control variables. $C'_{T,min}$, $C'_{T,max}$, ϕ_{min}, and ϕ_{max} are the lower and upper bounds of thrust coefficients and yaw angles, respectively. To obtain the optimal control policy for the problem (3)-(6), a model-free control framework based on the RL scheme will be introduced in the next section.

III. DN-DDPG Scheme for Wind Farm Control

In general, RL aims to provide data-driven solutions for the Markov Decision Process (MDP). MDP is commonly described by a transition $[s_t, a_t, r_t, s_{t+1}]$, where $s_t \in S$ is the current state at time t, $a_t \in A$ is the action, $r_t \in R$ is the reward, and $s_{t+1} \in S$ is the next state, and here S, A, and R denote the state, action and reward spaces, respectively. The goal of RL is to learn an effective policy $\pi(s) : S \rightarrow A$ by maximizing a long-time reward $R_T = \sum_t^{\infty} \gamma_t r_t(s_t, a_t)$, where $\gamma_t = \gamma^t$ with $\gamma \in (0, 1]$, and $r_t(s_t, a_t)$ indicates the instantaneous reward at s_t after taking the action a_t.

A. DDPG Method

Deep deterministic policy gradient (DDPG) [28] is a state-of-the-art deep RL algorithm. It extends the deep Q-network (DQN) [32] to the continuous and high-dimensional space [33]. Typically two kinds of neural networks are applied in DDPG based on the actor-critic structure - the main networks and the target networks. We denote the parameters of main critic and actor networks by θ^Q and θ^π respectively. Therein, the main critic is designed to approximate a so-called action-value function $Q(s, a|\theta^Q)$, which represents the expected long-term reward at the state s_t after executing action a_t and satisfies

$$Q(s_t, a_t) = r_t + \gamma Q(s_{t+1}, \pi(s_{t+1})) \tag{7}$$

The main actor aims to find an optimal control policy $\pi^*(s_t)$ to maximize $Q(s_t, a_t)$. The target critic and actor networks with parameters $\theta^{Q'}$ and $\theta^{\pi'}$ are used for improving the stability of the training process via tracking the their main counterparts. They are updated by the soft replacement strategy:

$$\begin{aligned} \theta^{Q'} &\leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'} \\ \theta^{\pi'} &\leftarrow \tau\theta^\pi + (1-\tau)\theta^{\pi'} \end{aligned} \tag{8}$$

where $\tau \in (0, 1]$ is a user-defined constant. The main critic is updated by minimizing the loss function

$$L = \frac{1}{b}\sum_{j=1}^{b} w_j z_j^2 \tag{9}$$

where b is the sampled batch size, w_j is the weight of the j-th sample, and z_j denotes the TD-errors described by

$$z_j = r_j + \gamma Q'(s_{j+1}, \pi'(s_{j+1}|\theta^{Q'})) - Q(s_j, a_j|\theta^Q) \tag{10}$$

The main actor is trained by the following policy gradient strategy

$$\nabla_{\theta^\pi} J \sim \frac{1}{N}\sum_{i=1}^{N} w_j[\nabla_{a_j} Q(s_j, a_j|\theta^Q)\nabla_{\theta^\pi}\pi(s_j|\theta^\pi)] \tag{11}$$

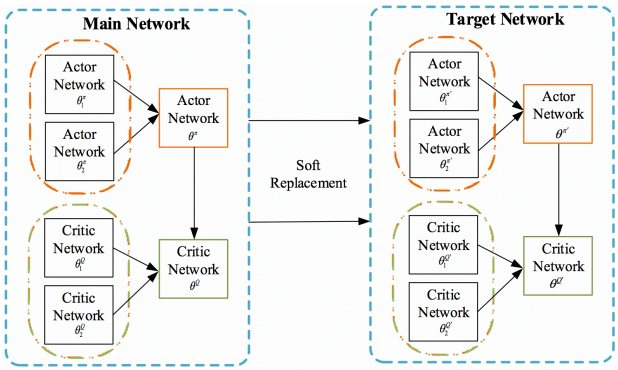


Fig. 1. The main structure of our DN-DDPG method.

ALGORITHM 1 DN-DDPG ALGORITHM

1 : Initialize the main critic networks $Q_1(s_1, a_1|\theta_1^Q)$ and $Q_2(s_2, a_2|\theta_2^Q)$, main actor networks $\pi_1(s_1|\theta_1^\pi)$ and $\pi_2(s_2|\theta_2^\pi)$, target critic networks $Q'_1(s_1, a_1|\theta_1^{Q'})$ and $Q'_2(s_2, a_2|\theta_2^{Q'})$, and target actor networks $\pi'_1(s_1|\theta_1^{\pi'})$ and $\pi'_2(s_2|\theta_2^{\pi'})$ with weights θ_1^Q, θ_2^Q, θ_1^π, θ_2^π, $\theta_1^{Q'}$, $\theta_2^{Q'}$, $\theta_1^{\pi'}$ and $\theta_2^{\pi'}$
2 : Initialize the memory buffer $\mathcal{M}$
3 : **for** $episode = 1$ *to* v **do**
4 : Receive the first observation state $s_{1,0}$ and $s_{2,0}$
5 : **for** $step = 1, ..., T_s$ **do**
6 : Select actions $a_{1,t} = \pi_1(s_{1,t}|\theta_1^\pi) + \mathcal{N}_t$ and $a_{2,t} = \pi_2(s_{2,t}|\theta_2^\pi) + \mathcal{N}_t$
7 : Execute the action $a_{1,t}$ and $a_{2,t}$, calculate the reward r_t, and observe the next state $s_{1,t+1}$ and $s_{2,t+1}$
8 : Store the transition $[s_{1,t}, s_{2,t}, a_{1,t}, a_{2,t}, r_t, s_{1,t+1}s_{2,t+1}]$ and its priorities into $\mathcal{M}$
9 : Sample a batch of b transitions according to the PER
10 : Compute the ISWs of each transition
11 : Update the priority of transitions based on the TD-errors
12 : Update the weights θ_1^Q and θ_2^Q of the main critic networks by minimizing the loss function shown in (15) and (16), respectively
13 : Update the weights θ_1^π and θ_2^π of the main actor networks by calculating the policy gradient shown in (17) and (18), respectively
14 : Update the weights $\theta_1^{Q'}$, $\theta_2^{Q'}$, $\theta_1^{\pi'}$, and $\theta_2^{\pi'}$ of the target critic and actor networks by soft replacement based on (13) and (14), respectively
15 : **end for**
16 : end for

During the training process, the policy is added with noise for exploration purposes. The transition $[s_j, a_j, r_j, s_{j+1}]$ will be stored into a memory buffer $\mathcal{M}$. In standard DDPG, an experience replay method is employed to sample transitions in a uniformly random way at every training step. Here we employ the prioritized experience replay (PER) [29] method

to prioritize transitions by their TD-errors and use those priorities for sampling. This can improve the training efficiency. Meanwhile, the bias induced by distribution mismatching is corrected via the importance-sampling weight (ISW) [34], i.e. w_j in (10).

B. Double-Network-Based DDPG

The standard DDPG method integrates all control inputs (i.e. yaw angles and thrust coefficients in this study) into one set of critic-actor networks. This design is feasible but may lead to performance degradation because the yaw angles and the thrust coefficients have distinctive and incompatible features. Specifically, yaw actuators usually have large time constants and cannot tolerate rapid yaw changes. While thrust coefficients typically have a much shorter response time than yaw angles and can adapt to irregular and rapid variations. Briefly speaking, the variation trend of yaw angles is slow while that of thrust coefficients is relatively rapid. When these two distinctive and incompatible control variables are integrated into one set of critic-actor networks, it is difficult for network parameters to be properly trained to meet such different variation trends. Therefore, different granularities are required in control policy learning, which is challenging for standard DDPG methods. We design a new RL algorithm to handle this issue and refer to it as the double network-based DDPG (DN-DDPG). The main framework of DN-DDPG is shown in Fig. 1.

Unlike the original DDPG method, the proposed DN-DDPG method employs two sets of critic-actor networks to evaluate yaw and thrust coefficient policies, respectively. Instead of updating the network parameters via a unified way like DDPG, the two sets of critic-actor networks in DN-DDPG are trained separately. Specifically, each set of networks has a main critic-actor structure and a target critic-actor structure, as defined in Sec. III.A and illustrated in Fig. 1. The original action a is divided into a_1 (for yaw angles) and a_2 (for thrust coefficients), which are trained by their corresponding actor networks separately. Each critic is employed to guide the training of the corresponding actor. Based on all these designs, the control actions of yaw angles and thrust coefficients are separately decided by their corresponding networks to meet different variation trends. To achieve the combination of yaw control and induction control, these two sets of critic-actor networks are linked by a central reward. That is to say, these two sets of critic-actor networks use the same reward to update their parameters, which serves as a link between the yaw control and the induction control.

As a short summary, our DN-DDPG can balance the yaw control and induction control. Its specially designed structure can handle the incompatibility between different control signals, providing a reliable training process and ensuring superior performance over standard DDPG methods.

The main actor networks and their parameters are presented by $\pi_1(s_{1,j}|\theta_1^\pi)$, $\pi_2(s_{2,j}|\theta_2^\pi)$, θ_1^π and θ_2^π, respectively. Also, the main critic networks and their parameters are denoted by $Q_1(s_{1,j}, a_{1,j}|\theta_1^Q)$, $Q_2(s_{2,j}, a_{2,j}|\theta_2^Q)$, θ_1^Q and θ_2^Q, respectively. Similar with the standard DDPG, the target networks are designed as the additional copies of main networks, which are employed to enhance the learning stability and are updated by the soft replacement strategy:

$$\begin{aligned}\theta_1^{Q'} &\leftarrow \tau\theta_1^Q + (1-\tau)\theta_1^{Q'} \\ \theta_1^{\pi'} &\leftarrow \tau\theta_1^\pi + (1-\tau)\theta_1^{\pi'}\end{aligned} \tag{12}$$

$$\begin{aligned}\theta_2^{Q'} &\leftarrow \tau\theta_2^Q + (1-\tau)\theta_2^{Q'} \\ \theta_2^{\pi'} &\leftarrow \tau\theta_2^\pi + (1-\tau)\theta_2^{\pi'}\end{aligned} \tag{13}$$

The main critics are updated by minimizing the loss functions based on TD-errors:

$$L_1 = \frac{1}{b}\sum_{j=1}^{b} w_j z_{1,j} \tag{14}$$

$$L_2 = \frac{1}{b}\sum_{j=1}^{b} w_j z_{2,j} \tag{15}$$

where $z_{1,j} = (r_j + \gamma Q_1'(s_{1,j+1}, \pi_1'(s_{1,j+1}|\theta_1^{Q'})) - Q_1(s_{1,j}, a_{1,j}|\theta_1^Q))^2$, and $z_{2,j} = (r_j + \gamma Q_2'(s_{2,j+1}, \pi_2'(s_{2,j+1}|\theta_2^{Q'})) - Q_2(s_{2,j}, a_{2,j}|\theta_2^Q))^2$. The main actors are updated by the policy gradient strategy

$$\nabla_{\theta_1^\pi} J_1 \sim \frac{1}{N}\sum_{i=1}^{N} w_j[\nabla_{a_{1,j}} Q_1(s_{1,j}, a_{1,j}|\theta_1^Q)\nabla_{\theta_1^\pi}\pi_1(s_{1,j}|\theta_1^\pi)] \tag{16}$$

$$\nabla_{\theta_2^\pi} J_2 \sim \frac{1}{N}\sum_{i=1}^{N} w_j[\nabla_{a_{2,j}} Q_2(s_{2,j}, a_{2,j}|\theta_2^Q)\nabla_{\theta_2^\pi}\pi_2(s_{2,j}|\theta_2^\pi)] \tag{17}$$

The transitions $[s_{1,t}, s_{2,t}, a_{1,t}, a_{2,t}, r_t, s_{1,t+1}, s_{2,t+1}]$ are stored in the memory buffer and sampled via PER in learning steps. The detailed application procedure of our DN-DDPG is presented in Algorithm 1. In addition, to clearly show the difference between DDPG and our DN-DDPG, their architectures are demonstrated in Fig. 3 and Fig. 4, respectively.

C. Application to a dynamic wind farm simulator WFSim

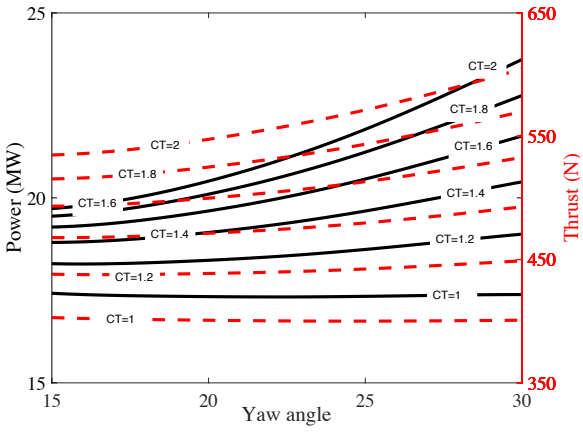
In order to validate the effectiveness of our DN-DDPG method, a dynamic wind farm simulator (WFSim) [30] is employed. WFSim establishes the flow field via the 2-dimensional Navier-Stokes equations:

$$\rho(\frac{\partial u}{\partial t} + \nabla\cdot u\mathbf{u}) = -\frac{\partial p}{\partial x} + \nabla(\sigma grad(u)) + f_x + T_x \tag{18}$$

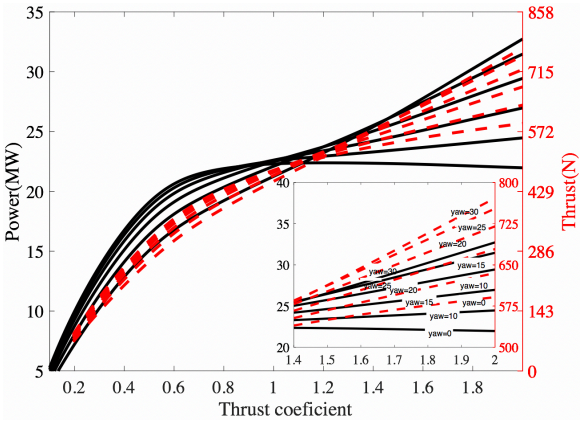
$$\rho(\frac{\partial v}{\partial t} + \nabla\cdot v\mathbf{u}) = -\frac{\partial p}{\partial y} + \nabla(\sigma grad(v)) + f_y \tag{19}$$

$$\rho grad(\mathbf{u}) = 0 \tag{20}$$

where ρ is the air density, p is the pressure, and σ is the dynamic viscosity. Also, $\mathbf{u} = [u, v]^T$ with u and v represent the velocities in the x and y directions, respectively. The operations $grad$, ∇, and $\frac{\partial}{\partial\cdot}$ denote the gradient, divergence, and partial derivative, respectively. In addition, f_x and f_y



(a)



(b)

Fig. 2. The relationship between power and thrust with respect to control variables. (a) The relationship between power and thrust with respect to yaw angle. (b) The relationship between power, thrust with respect to thrust coefficient.

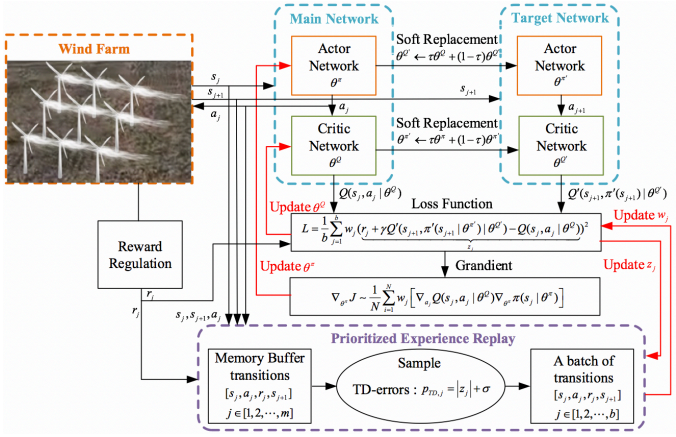


Fig. 3. The architecture of the DDPG.

denote the forces exerted by turbines in the x and y directions. T_x is the turbulence model.

Instead of directly employing the farm's total power output as reward, some modifications are made from the standpoint of practical engineering. As shown in Fig. 2 (where the initial longitude velocity is set as 10m/s), the turbines' powers/thrusts have strong correlations with their control variables. From Fig. 2(a), it is can be observed that the thrust raises with an increased yaw angle, which may cause heavy structural loads and lifetime degradation of infrastructure. Fig. 2(b) indicates that the state space with small thrust coefficients are meaningless for exploration, because the induced power outputs are quite small. Therefore, to avoid large structural loads and facilitate the learning process, we design the one-step reward as follows

$$r_t = k_p \sum_{i=1}^{N} P_i - k_f \sum_{i=1}^{N} \phi_i + k_c \sum_{i=1}^{N} C'_{T_i} \tag{21}$$

where k_p, k_f, k_c are the user-defined constant for scaling. The second term in (21) is a penalty term for large yaw offsets. The third term in (21) aims to weaken the possibility of searching the region with small thrust coefficients. By further taking the physical limits of thrust coefficient and yaw angle into account, a constrained control policy can be obtained to maximize a long-time reward $R = \sum_t^{\infty} \gamma_t r_t(s_t, a_t)$.

IV. Numerical Simulations

A. Simulation results based on WFSim

In this section, simulation results with WFSim are illustrated to show the effectiveness of our DN-DDPG algorithm.

A nine-turbine wind farm is employed. The flow field domain is set as $2519m \times 1558m$, which is divided into a spatial grid with 100×42 cells for numerical simulation. The air density is $\rho = 1.2kg/m^3$. The rotor diameter of each turbine is $126.4m$. The initial longitude velocity and lateral speeds are selected as $10m/s$ and $0m/s$.

Both the standard DDPG and our DN-DDPG are employed to carry out simulations. During the training process, the sizes of episode, step, memory buffer, and batch are selected as $\upsilon = 200$, $t_s = 200$, $\mathcal{M} = 10000$, and $b = 128$, respectively. The discount factor is set to $\gamma = 0.95$, and the constant for soft replacement is $\tau = 0.01$. The yaw angle ϕ_i and thrust coefficient C'_{T_i} are constrained by $0° \leq \phi_i \leq 30°$ and $0.1 \leq C'_{T_i} \leq 2$, respectively. The yaw angle variation rate is chosen as $0° \leq \Delta\phi_i \leq 1°$ and the rate of thrust coefficient is selected as $0 \leq \Delta C'_{T_i} \leq 0.2$. The weights in the reward function are $k_p = 1$, $k_f = 0.5$, and $k_c = 0.2$.

The initial flow field is shown in Fig. 5(a). The black vertical segments denote the nine turbines. T_i indicates the i-th turbine, and P_i (MW) denotes the power produced by the i-th turbine. R_i and C_i stand for the i-th row and column, respectively. After the training is completed, the resulting wind flow field at time $T = 700s$ under the DN-DDPG method is shown in Fig. 5(b). The power production for each turbine is marked for better readability. Besides, the control variables of each turbine are given in Fig. 6. One can see that thrust coefficients and yaw angles are all within their limits. To enhance the power production and avoid large loads, the trust coefficients are always larger than 1 and most yaw angles do not reach their upper bounds. This result meets our expectation as discussed in Sec. III.C. Moreover, the yaw angles in the first row are relatively larger than that in other rows such that less power is captured in the first row and more wind flow is propagated to

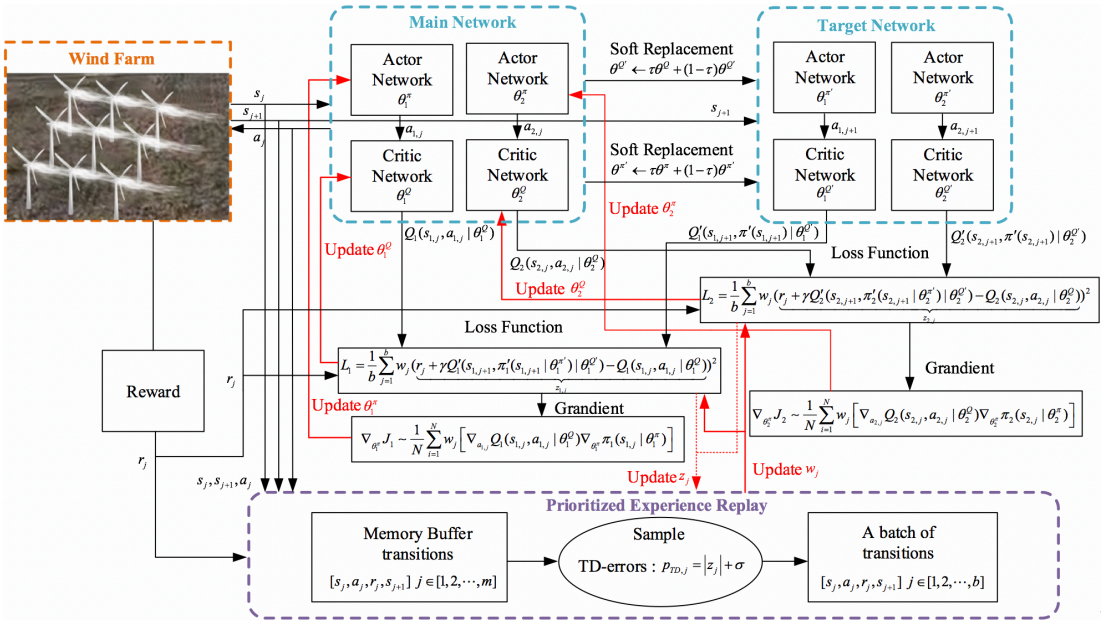


Fig. 4. The architecture of the DN-DDPG.

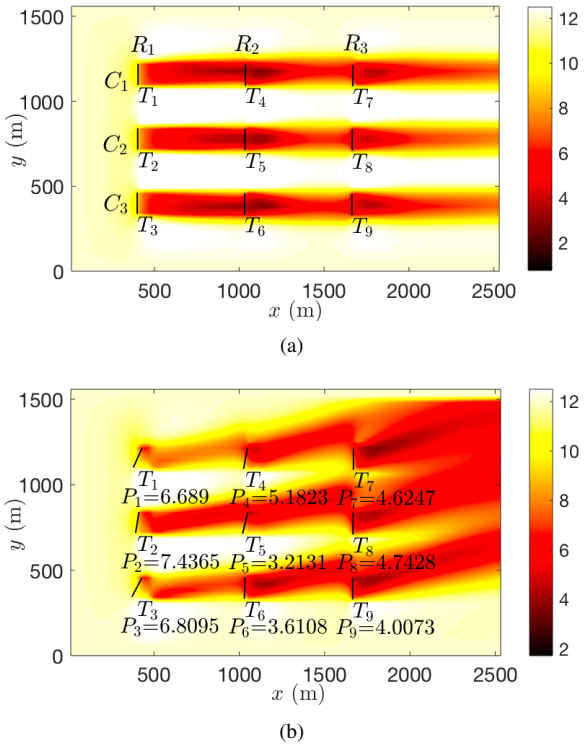


Fig. 5. Nine-turbine flow field. (a) Flow field under the greedy strategy. (b) Flow field under the proposed DN-DDPG strategy.

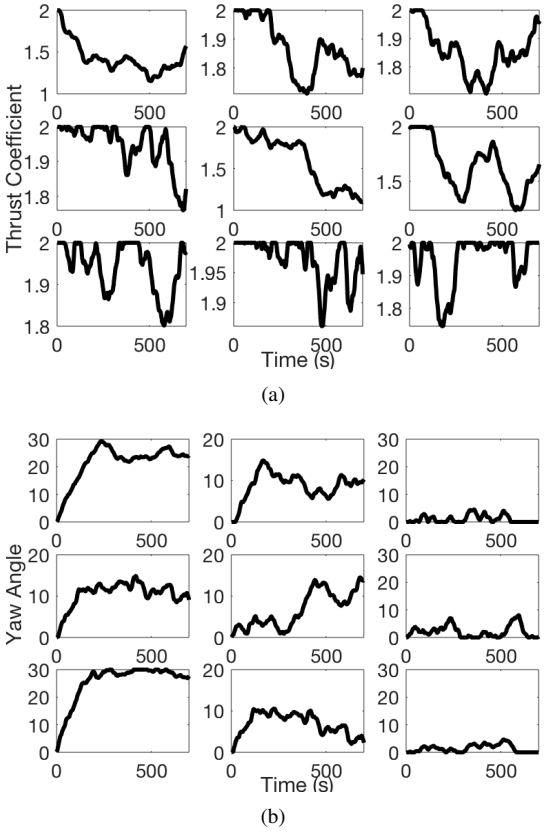


Fig. 6. Control signals of all turbines. (a) Thrust coefficients. (b) Yaw angles.

the downstream field. Meanwhile, the yaw angles in the last row are small and near zero in the steady condition to capture more power. Another observation is that the yaw angle changes gradually which is distinctive from the thrust coefficient that varies irregularly and changes fast in several short slots. This desirable behavior coincides with the results in [10], [15].

To compare the performance between DDPG and our DN-DDPG, the normalized cumulative rewards are shown in Fig. 7. It clearly indicates that the DN-DDPG method reaches larger cumulative rewards than the DDPG method. In addition, the results under the greedy control strategy using the Betz-optimal thrust coefficient $C'_{T_i} = 2$ and yaw angle $\phi_i = 0°$ are

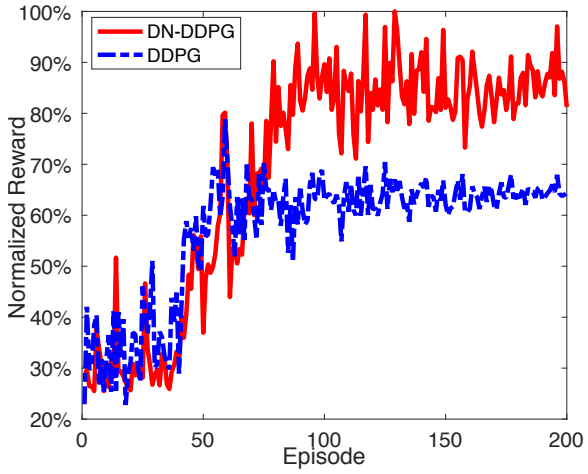


Fig. 7. Normalized cumulative rewards under different methods.

presented for comparison in Fig. 8. It can be observed that DN-DDPG outperforms DDPG by increasing the power generation to approximately 33% compared to 26% by the original DDPG method. Note that the normalized power production P_N is calculated using the total power P_{all} under DDPG and DN-DDPG and the greedy result P_{greedy} as follows

$$P_N = \frac{P_{all}}{P_{greedy}} \times 100\% \tag{22}$$

Fig. 8(b) depicts the power production of each row normalized by the power of the first row in the greedy case. Compared with the greedy control strategy, DDPG and DN-DDPG decrease the power in the first row by 8% but increase that nearly 30% in the downstream turbines. As a consequence, the total power is improved than the greedy case. Furthermore, the wind farm power efficiency η of the DN-DDPG and DDPG methods are calculated by

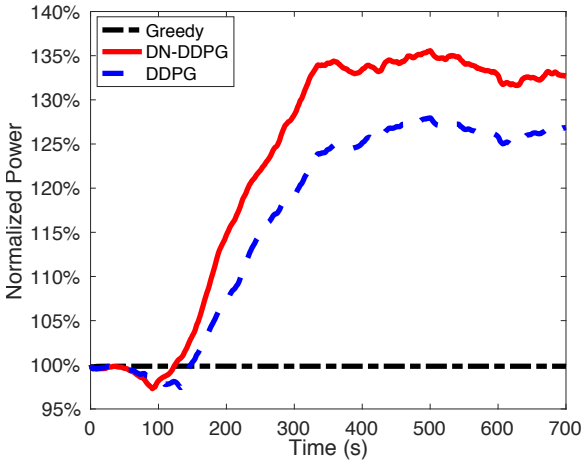
$$\eta = \frac{P_{all}}{N \cdot \overline{P}_{R_1}} \tag{23}$$

where $\overline{P}_{R_1}$ denotes the averaged first-row power and N is the number of turbines. According to (23), the power efficiency of the designed DN-DDPG method and the original DDPG method is 32% and 29%, respectively.

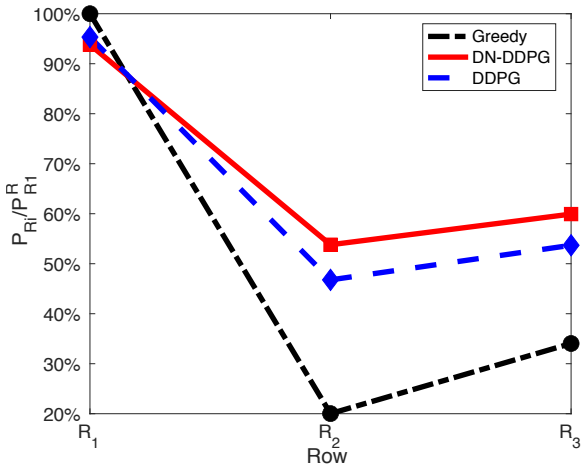
B. Further validation under different scenarios

1) Simulation results under different wind speeds: Consider the initial wind speeds varying from $8m/s$ to $14m/s$, the power production under our DN-DDPG algorithm compared with the greedy control are given in Fig. 9. It can be observed that the DN-DDPG based control policy can generate more power than the greedy control strategy under all wind speeds. Therefore, the proposed method is preferable in creating more power under a large wind-speed range.

2) Simulation results under a different wind-farm layout: To further verify the effectiveness of the proposed DN-DDPG method, a different wind-farm layout is employed in simulations, as shown in Fig. 10(a) and Fig. 10(b). Therein, the turbine positions in the second and last rows are varied at Y direction by 0.5 of the rotor diameter. The normalized power generations are illustrated in Fig. 11(a). Compared with the greedy strategy, the power production under DN-DDPG in this



(a)



(b)

Fig. 8. Comparison with the greedy control strategy. (a) Normalized total power production. (b) Normalized power production of each row.

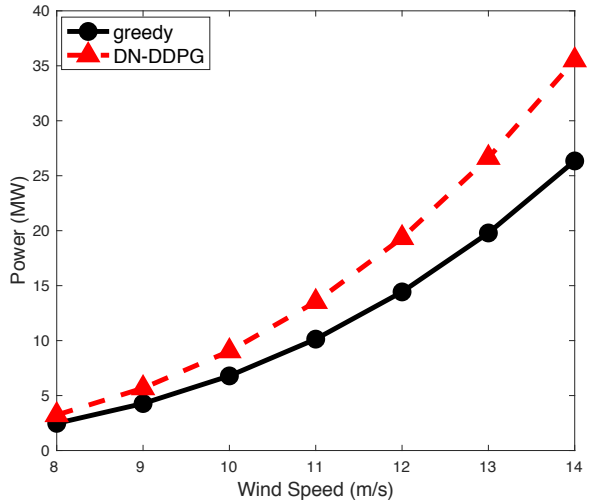


Fig. 9. Simulation results of the power outputs under different inflow wind speeds.

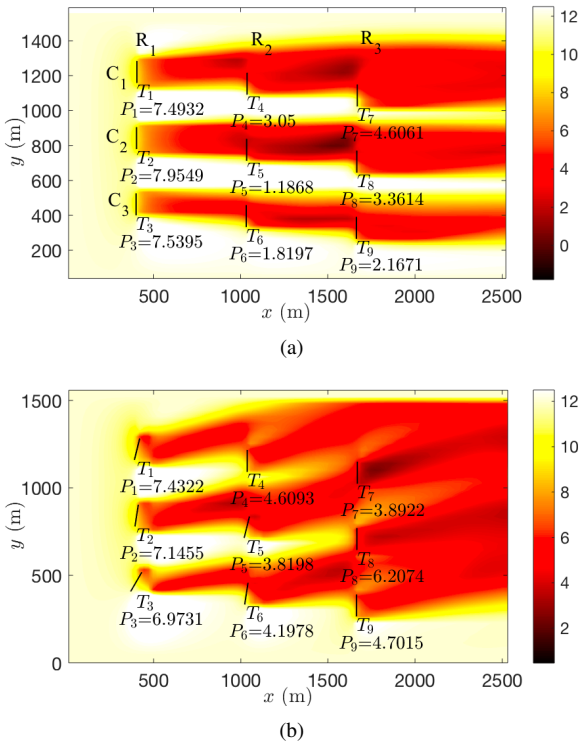


Fig. 10. Simulation results of the flow field. (a) Flow field under the greedy strategy. (b) Flow field under the proposed DN-DDPG strategy.

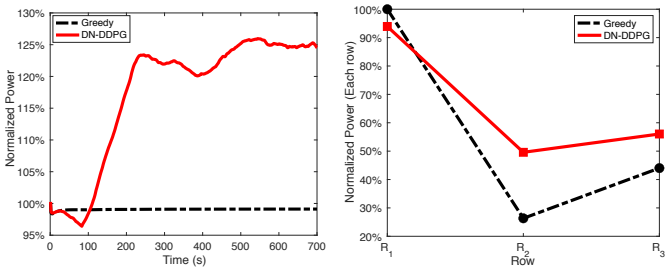


Fig. 11. Simulation results of the power production via the proposed DN-DDPG method and the greedy control strategy. (a) The total power production. (b) The power production for each row.

scenario is improved by 25%. The power production of each row normalized by the power of the first row under the greedy control is displayed in Fig. 11(b). Similar to the analysis for Fig. 8(b), DN-DDPG decreases the power in the first row but leads to more power outputs in the downstream turbines. Therefore, the total power generation is improved.

3) Simulation results under turbulence intensity uncertainties and wind speed uncertainties: The turbulence intensity uncertainties and wind speed uncertainties are considered in this subsection to further validate the adaptability and robustness of the proposed method. We carry out Monte Carlo simulations to this end. Note that the turbulence in WFSim is modelled by the Prandtl's mixing length model [30]. Particularly, the tuning variables (l_s, d, d') decide the turbulence intensity, and these parameters are considered to be uncertain and unknown for the proposed DN-DDPG method. The ranges

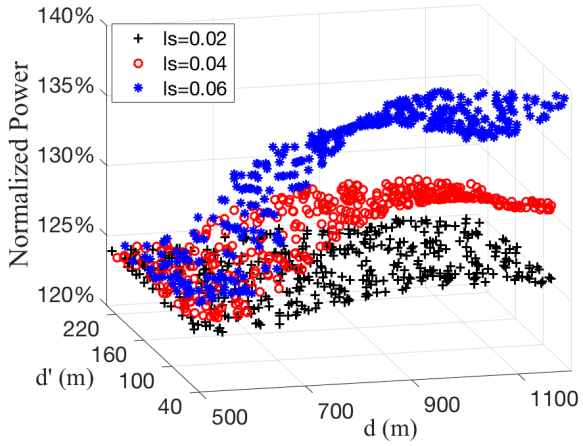


Fig. 12. Monte Carlo simulation results under turbulence intensity uncertainties and wind speed uncertainties.

of d and d' are set to be $d \in [500, 1200]m$ and $d' \in [40, 240]m$ in Monte Carlo simulations, and the initial wind speed is randomly selected from 9m/s to 11m/s. Moreover, simulation cases with different l_s $(0.02, 0.04,$ and $0.06)$ are considered. In each case, a set of 500 Monte Carlo simulations with randomly selected uncertain parameters (e.g. d, d', and the wind speed) is carried out to test the proposed DN-DDPG method. Simulation results of the normalized power generation w.r.t the power output under the greedy strategy is provided in Fig. 12. It illustrates that the proposed DN-DDPG method can still achieve clear power generation increases under uncertain turbulence intensity and wind speed. An approximate 24% increase on the power production is obtained even in the extreme case. These results show that our method successfully adapts to turbulence-intensity and wind-speed uncertainties.

4) Simulation results of different methods: In order to further evaluate the performance of the proposed DN-DDPG method, the nonlinear model predictive control (NMPC) controller in [16] is employed to make comparison. This most recent wind farm control strategy has been proven to have strong optimizing and constraint handling abilities. Simulation results of the normalized power production under the NMPC method, DDPG method, greedy control strategy and our DN-DDPG method are shown in Fig. 13. These results indicate that DN-DDPG has the best performance among all the methods, which can significantly increase the power generation by 33% on average compared with the greedy strategy. In contrast, that of NMPC and DDPG are 18% and 26%, respectively. In addition, the simulation results of the normalized power production of each row in the farm are presented in Fig. 14. Compared with the greedy control, the DN-DDPG, DDPG and NMPC methods decrease the power in the first row but clearly increase the power in the second and third row.

To sum up, all these simulation results indicate that the proposed DN-DDPG method is effective and feasible, and it leads to superior performance than the greedy strategy, the DDPG algorithm and the NMPC method.

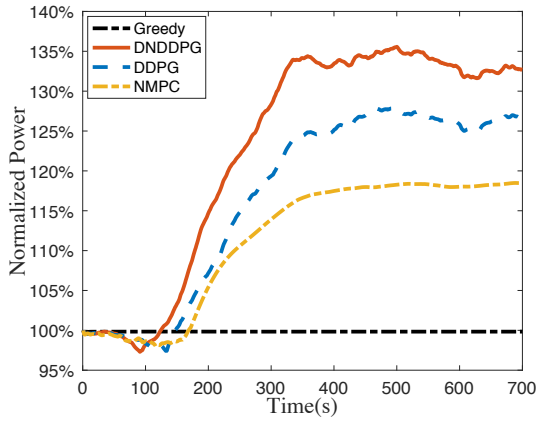


Fig. 13. Normalized power production under different methods.

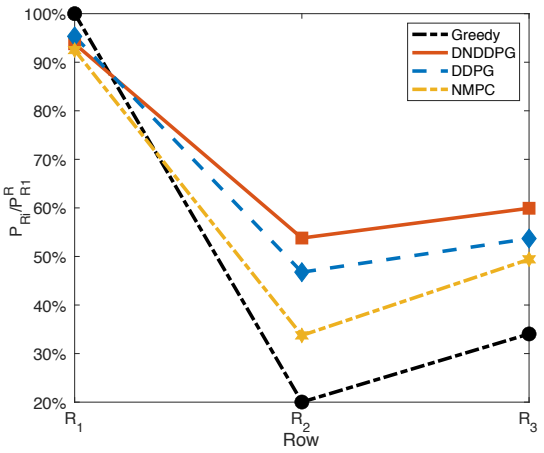


Fig. 14. Normalized power production of each row under different methods.

V. Conclusion

A double-network-based deep deterministic policy gradient (DN-DDPG) algorithm was proposed in this work to provide a data-driven model-free solution for wind-farm power maximization problems. It was developed by constructing two sets of critic-actor networks such that the induction and yaw control policies can be generated simultaneously and separately. It can handle the incompatibility between different control signals (thrust coefficients and yaw angles), providing a more reliable training process and ensuring superior performance over the standard DDPG algorithm. Simulation results with a dynamic wind farm simulator verified the effectiveness and adaptability of the proposed DN-DDPG algorithm. Although the proposed DN-DDPG method can significantly increase the power production of the wind farm, its computational complexity is higher than the conventional greedy control strategy. Therefore, future research will devote to the improvement over the learning efficiency and effectiveness of the DN-DDPG.

References

[1] X. Luo, J. Sun, L. Wang, W. Wang, W. Zhao, J. Wu, J. Zhang, and Z. Zhang, "Short-Term Wind Speed Forecasting via Stacked Extreme Learning Machine With Generalized Correntropy," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 11, pp. 4963-4971, Nov. 2018.

[2] R. Hidalgo-Leon, J. Urquizo, J. Macias, D. Siguenza, P. Singh, J. Wu, and G. Soriano, "Energy Harvesting Technologies: Analysis of their potential for supplying power to sensors in buildings," *in Proc. 2018 IEEE Third Ecuador Technical Chapters Meeting (ETCM)*, pp. 1-6, Oct. 2018.

[3] R. Hidalgo-Leon, J. Urquizo, J. Litardo, Y. Munoz-Jadan, P. Singh and J. Wu, "Simulation of battery discharge emulator using power electronics device with cascaded P-I control," *in Proc. 2020 IEEE International Conference on Industrial Technology (ICIT)*, pp. 959-964, Feb. 2020.

[4] R. Hidalgo-Leon, D. Siguenza, C. Sanchez, J. Leon, P. Jacome-Ruiz, J. Wu, and D. Ortiz, "A survey of battery energy storage system (BESS), applications and environmental impacts in power systems," *in Proc. 2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM)*, pp. 1-6, Oct. 2017.

[5] H. Zhao, Q. Wu, Q. Guo, H. Sun, and Y. Xue, "Distributed model predictive control of a wind farm for optimal active power control part II: Implementation with clustering-based piece-wise affine wind turbine model," *IEEE Transactions on Sustainable Energy*, vol. 6, no. 3, pp. 840-849, Apr. 2015.

[6] C. Zhang, H., Chen, K. Shi, Z. Liang, W. Mo, and D. Hua, "A multi-time reactive power optimization under interval uncertainty of renewable power generation by an interval sequential quadratic programming method," *IEEE Transactions on Sustainable Energy*, vol. 10, no. 3, pp. 1086-1097, Jul. 2018.

[7] Y. Wang, H. Liu, H. Long, Z. Zhang, and S. Yang, "Differential evolution with a new encoding mechanism for optimizing wind farm layout". *IEEE Transactions on Industrial Informatics*, vol. 14, no. 3, pp.1040-1054, Sep. 2017.

[8] G. P. Corten, and P. Schaak, "Heat and flux: Increase of wind farm production by reduction of the axial induction," *in Proceedings of the European Wind Energy Conference*, Jun. 2003.

[9] Z. Dar, K. Kar, O. Sahni, and J. H. Chow, "Windfarm power optimization using yaw angle control," *IEEE Transactions on Sustainable Energy*, vol. 8, no. 1, pp. 104-116, Jun. 2016.

[10] W. Munters, and J. Meyers, "Dynamic strategies for yaw and induction control of wind farms based on large-eddy simulation and optimization," *Energies*, vol. 11, no. 1, pp. 177, Jan. 2018.

[11] J. Lee, E. Son, B. Hwang, and S. Lee, "Blade pitch angle control for aerodynamic performance optimization of a wind farm," *Renewable energy*, vol. 54, pp. 124-130, Jun. 2013.

[12] B.Dou, T. Qu, L. Lei, and P. Zeng, "Optimization of wind turbine yaw angles in a wind farm using a three-dimensional yawed wake model," *Energy*, vol. 209, no. 15, p. 118415, Oct. 2020.

[13] D. van der Hoek, S. Kanev, J. Allin, D. Bieniek, and N. Mittelmeier, "Effects of axial induction control on wind farm energy production-A field test," *Renewable Energy*, vol. 140, pp. 994-1003, Sep. 2019.

[14] J. P. Goit, and J. Meyers, "Optimal control of energy extraction in wind-farm boundary layers," *Journal of Fluid Mechanics*, vol. 768, pp. 5-50, Feb. 2015.

[15] W. Munters, and J. Meyers, J. "An optimal control framework for dynamic induction control of wind farms and their interaction with the atmospheric boundary layer," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 375, no. 2091, p. 20160100, Mar. 2017.

[16] M. Vali, V. Petrovic, S. Boersma, J. W. van Wingerden, and M. Kuhn, "Adjoint-based model predictive control of wind farms: Beyond the quasi steady-state power maximization," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 4510-4515, Jul. 2017.

[17] J. Barreiro-Gomez, C. Ocampo-Martinez, F. Bianchi, and N. Quijano, "Model-free control for wind farms using a gradient estimation-based algorithm," *in European Control Conference*, pp. 1516-1521, Jul. 2015.

[18] J. R. Marden, S. D. Ruben, and L. Y. Pao, "A model-free approach to wind farm control using game theoretic methods," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1207-1214, Jul. 2013.

[19] J. Park, and K. H. Law, "A data-driven, cooperative wind farm control to maximize the total power production," *Applied Energy*, vol. 165, pp. 151-165, Mar. 2016.

[20] M. A. Ahmad, M. R. Hao, R. M. T. R. Ismail, and A. N. K. Nasir, "Model-free wind farm control based on random search," *in IEEE International Conference on Automatic Control and Intelligent Systems*, pp. 131-134, Oct. 2016.

[21] M. A. Ahmad, S. I. Azuma, and T. Sugie, "A model-free approach for maximizing power production of wind farm using multi-resolution simultaneous perturbation stochastic approximation," *Energies*, vol. 7, no. 9, pp. 5624-5646, Aug. 2014.

[22] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *The International Journal of Robotics Research*, vol. 32, no. 11, pp.1238-1274, Aug. 2013.
[23] D. Isele, R. Rahimi, A. Cosgun, K. Subramanian, and K. Fujimura, "Navigating occluded intersections with autonomous vehicles using deep reinforcement learning," *in IEEE International Conference on Robotics and Automation*, pp. 2034-2039, May. 2018.
[24] X. Wang, P. Shi, C. Wen, and Y. Zhao, "Design of Parameter-self-tuning Controller Based on Reinforcement Learning for Tracking Non-cooperative Targets in Space," *IEEE Transactions on Aerospace and Electronic Systems*, Apr. 2020.
[25] P. Stanfel, K. Johnson, C. J. Bay, and J. King, "A Distributed Reinforcement Learning Yaw Control Approach for Wind Farm Energy Capture Maximization," *in American Control Conference*, pp. 4065-4070, Jul. 2020.
[26] A. Saenz-Aguirre, E. Zulueta, U. Fernandez-Gamiz, J. Lozano, and J. M. Lopez-Guede, "Artificial neural network based reinforcement learning for wind turbine yaw control," *Energies*, vol. 12, no. 3, pp.436. Jan. 2019.
[27] H. Zhao, J. Zhao, J. Qiu, G. Liang, and Z. Y. Dong, "Cooperative Wind Farm Control with Deep Reinforcement Learning and Knowledge Assisted Learning," *IEEE Transactions on Industrial Informatics*, vol.16, no. 11, pp. 6912-6921, Feb. 2020.
[28] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D.Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," in International Conference on Machine Learning, 2016.
[29] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized experience replay," in International Conference on Machine Learning, 2016.
[30] S. Boersma, B. Doekemeijer, M. Vali, J. Meyers, and J. W. van Wingerden, "A control-oriented dynamic wind farm model: WFSim," *Wind Energy Science*, vol. 3, no. 1, pp.75-95, Mar. 2018.
[31] J. Park, and K. H. Law, "Cooperative wind turbine control for maximizing wind farm power using sequential convex programming," *Energy Conversion and Management*, vol. 101, pp. 295-316, Sep. 2015.
[32] E. Duryea, M. Ganger, and W. Hu, "Exploring deep reinforcement learning with multi q-learning," *Intelligent Control and Automation*, vol. 7, no. 4, pp.129-144, Nov. 2016.
[33] J. Duan, D. Shi, R. Diao, H. Li, Z. Wang, B. Zhang, D. Bian, and Z. Yi, "Deep-reinforcement-learning-based autonomous voltage control for power grid operations," *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp.814-817, Sep. 2019.
[34] A. R. Mahmood, H. P. van Hasselt, and R. S. Sutton, "Weighted importance sampling for off-policy learning with linear function approximation," *Advances in Neural Information Processing Systems*, pp. 3014-3022, 2014.

Jingjie Xie is currently working toward the Ph.D. degree in Engineering at the School of Engineering, University of Warwick, Coventry, UK. She received the B.S. degree in information engineering from Northwestern Polytechnical University, Xian, China, in 2016, and the M.S. degree in control science and engineering from Beijing University of Aeronautics and Astronautics, Beijing, China, in 2019. Her current research interests include reinforcement learning, deep learning, and intelligent control.

Hongyang Dong is currently a Research Fellow in Machine Learning and Intelligent Control at the School of Engineering, University of Warwick, Coventry, UK. He obtained his Ph.D. degree in Control Science and Engineering from Harbin Institute of Technology, Harbin, China, in 2018. His current research interests include reinforcement learning, intelligent control, adaptive control, and their applications.

Xiaowei Zhao is Professor of Control Engineering and an EPSRC Fellow at the School of Engineering, University of Warwick, Coventry, UK. He obtained the PhD degree in Control Theory from Imperial College London in 2010. After that he worked as a postdoctoral researcher at the University of Oxford for three years before joining Warwick in 2013. His main research areas are control theory with applications on offshore renewable energy systems, local smart energy systems, and autonomous systems.

Aris Karcanias is the Global Head of Energy & Utilities with PA Consulting, London, U.K. He has a technical engineering background, with extensive experience in leading board-level strategy, technology development, and M&A for leading utilities, manufacturers, and investors. His research interests include clean energy systems, their integration, and the redesign of the future energy system.